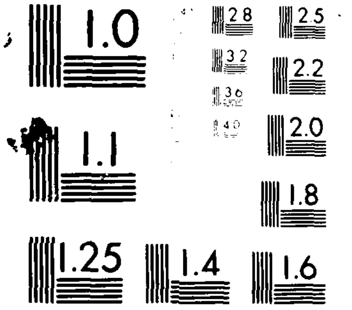
AD-A112 690 DAVID W TAYLOR NAVAL SHIP RESEARCH AND DEVELOPMENT CE--ETC F/G 15/5
SHIP ON-LINE SCHEDULER (SOS) USER'S MANUAL.(U)
MAR 82 R MELTON
UNCLASSIFIED DTNSRDC-82/009 NL

1 of 1
AD
A112690

END
DATE
FILMED
4 82
DTIC


1.0
2.8
2.5
3.2
2.2
3.6
4.0
2.0
1.1
1.8
1.25
1.4
1.6

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER
DTNSRDC-82/009

2. GOVT ACCESSION NO.
AD-A112 69Ø

3. RECIPIENT'S CATALOG NUMBER

4. TITLE (and Subtitle)
SHIP ON-LINE SCHEDULES (SOS) USER's MANUAL

5. TYPE OF REPORT & PERIOD COVERED
Final

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s)
R. Melton

8. CONTRACT OR GRANT NUMBER(s)

9. PERFORMING ORGANIZATION NAME AND ADDRESS
David Taylor Naval Ship Research and
Development Center
Bethesda, Maryland 20084

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
1870-041
OM&N Funds

11. CONTROLLING OFFICE NAME AND ADDRESS

12. REPORT DATE
March 1982

13. NUMBER OF PAGES
33

14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)

15. SECURITY CLASS. (of this report)
UNCLASSIFIED

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

APPROVED FOR PUBLIC RELEASE: DISTRIBUTION UNLIMITED

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Ships
Schedules
Cargo Requirements
Shipping
Ports
Berths
Queues

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Ship On-line Scheduler (SOS) is a software package designed to assist in scheduling cargo delivery among ports in a Navy supply environment. SOS is written in FORTRAN IV and is designed to run on the CDC 6000 series computers at DTNSRDC.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

Ship On-line Scheduler (SOS) is a software package designed to assist in scheduling cargo delivery among ports in a Navy supply environment. SOS is written in FORTRAN IV and is designed to run on the CDC 6000 series computers at DTNSRDC.

ADMINISTRATIVE INFORMATION

This sub-task was funded by the Maritime Affairs and Support Plans Branch (OP-405) with OM&N funds. The work was performed by the Logistics Planning Group (Code 1871) in the Computation, Mathematics, and Logistics Department of the David Taylor Naval Ship Research and Development Center, Carderock, Maryland under Work Unit 1870-041.

INTRODUCTION

The Ship On-Line Scheduler (SOS) is a software package designed to assist in scheduling cargo deliveries among ports in a Navy supply environment. The scheduling algorithm used by SOS requires a number of operational parameters, some of which are built into the program coding and some supplied by input. These parameters include ports, a port distance table which gives the distance in nautical miles between ports, ships and ship types characteristics, and cargo. The port characteristics, distance table, and ship type characteristics, are defined in the coding of SOS. Shipping resources and cargo information are entered as input.

SOS can consider up to 30 ports. For each port used in SOS, ship entry and departure times and cargo loading and unloading rates must be given. A descriptive name for each port must also be specified.

Deliveries and/or pickups of cargo are made by as many as 50 ships of the following 14 general ship types: BBS, BBF, ROROS, ROROF, SSCS, SSCF, NSSCS, NSSCF, COMBS, COMBF, LASH, STRHV, STRNSS, and SEABEE. Ship types are distinquished by their operational characteristics such as speed, loading time and manner of loading, and by the type of cargo they can carry and the ports they can service.

Cargo information is given in the form of orders, that is, a statement of quantity (mt's), type (1 to 9), and origin and destination ports. Up to 200 orders may be considered by SOS.

SOS uses a maximum vehicle utilization/least time algorithm. Since the order of load pick up and delivery is a function of the ship type selected, "first on, first off" and "first on, last off" procedures were incorporated in the algorithm. Routes are built to "maximum" efficiency, with the most efficient ship types selected for handling loads.

For SOS to be successful, the program must be easily usable by management personnel who have had minimal computer training. In addition, the scheduling program must execute rapidly to assure fast response to orders. For these reasons the SOS uses a packed linked list method of accessing and building ship schedules. Program procedures, execution instructions, and output file storage are simple.

SOS - CARGO SCHEDULING

Cargo scheduling takes place in four phases. The first phase is interactive data entry from a remote terminal. In the second phase (subroutine SOSN1) the program examines the input orders individually and sorts them to reduce ship order selection time. In the third phase (subroutine ROUTE) the ship type order lists are asembled into ship routes. The last phase converts the ship route arrays into usable printout (subroutine TCARP). Figure 1 gives the SOS System Flowchart.

METHOD and ALGORITHM

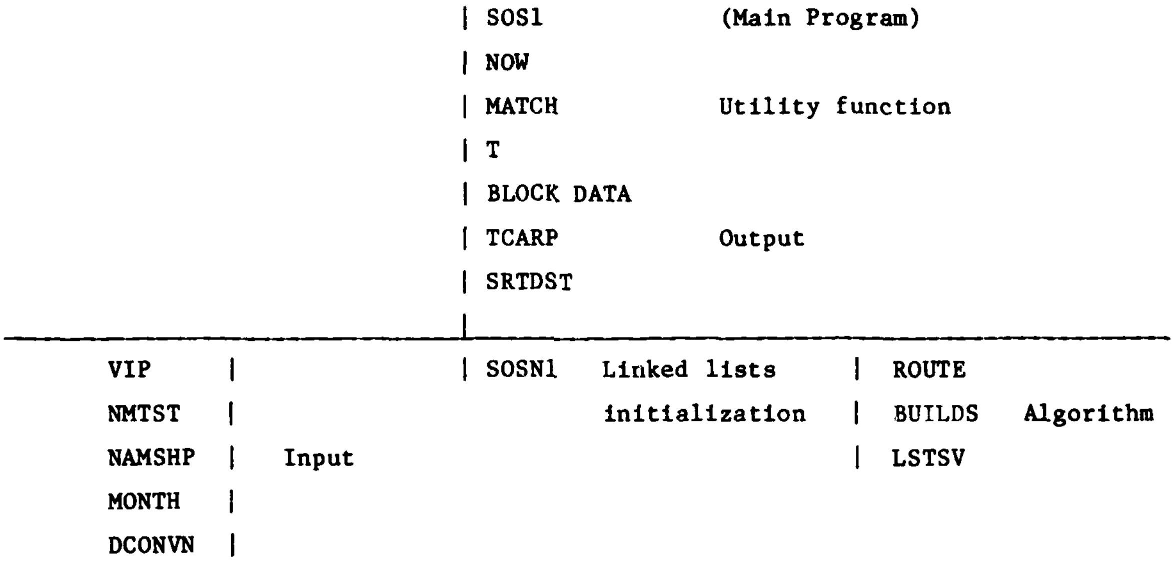
Route building is accomplished by subroutines ROUTE and BUILDS. The schedules for most efficient ship types are built first. This sequence of schedule building may be changed to fit the needs of the user.

The algorithm operates on the sorted list of orders. This list of orders is scanned to determine the combination of orders which will, if serviced by a single ship, provide the greatest time savings (or least time cost) over the situation in which each order is serviced by a separate ship. There is almost always a time savings involved in joining two or more orders in this manner. However, to prevent excessive order joining and over-utilization of individual ships, a least time savings restriction was added to the algorithm. Since joined order routes are assigned to preferred ship types first, the least time saving restriction reduces the number of joined orders and allows the assignment of single order routes to all available ships. If a minimum load requirement for the ship is not met, order segments will not be joined.

Having selected the best set of orders to start a ship's route, the algorithm examines the remaining orders in the list for that single order which, if joined to the route, results in the least time cost over servicing the order separately. As in the starting case the limit on time cost applies. The new order is placed at the end of the existing route, since examining intermediate positions along the route would be too time-consuming and the coding would be too complex.

The algorithm continues in this manner, adding orders to the end of the previous route, until the route time limit for the ship precludes further additions, or until the pool of unassigned orders is exhausted. In the former case the next ship of the same type is selected using the same method. In the latter case the algorithm proceeds directly to consideration of the next ship type.

The berthing delays and queuing at ports are determined randomly by a facility availability function. Parameters used by this function are set in the program coding but may be changed for each set of ports considered.



```
                              | SOS1          (Main Program)
                              | NOW
                              | MATCH         Utility function
                              | T
                              | BLOCK DATA
                              | TCARP         Output
                              | SRTDST
                              |
------------------------------------------------------------------
     VIP     |                | SOSN1  Linked lists     | ROUTE
     NMTST   |                       initialization     | BUILDS   Algorithm
     NAMSHP  |   Input                                  | LSTSV
     MONTH   |
     DCONVN  |
```

Figure 1 - SOS System Flochart

SPECIAL TECHNIQUES

Several techniques were used in the SOS program to reduce execution time and core requirements. The penalty for the gain in efficiency of the program is increased program complexi·y. These techniques are described here to help the user in understanding the program coding. One of these techniques has to do with the calculation of travel times between ports; two are general data storage techniques used to reduce sort times in the SOS algorithm.

TRAVEL TIME

The SOS program was designed to service up to 30 ports, and the algorithm makes frequent use of the travel times between ports. These times differ for the fourteen ship types, giving more than 12,600 intra-activity time measurements. The prohibitive cost of storing such a collection of data demands that this figure be reduced to a more manageable level; it is this problem that this technique addresses. The reduction in array size is based on an assumption of symmetry in the travel time matrix; i.e., the time to travel from port A to port B is the same as the time to travel from port B to port A. This assumption is justified by actual travel time data collected for the REACT* study. (Ship speed and distance traveled replace individual time measurements).

Applying this technique reduces the array from more than 12,600 storage locations to 420 with only a slight increase in the procedural code generated and with little or no decrease in the accuracy of the schedules.

LINKED LIST

The linked list method of data storage is demonstrably faster than the sorting method and uses considerably less core than the duplicate arrays method, both of which are discussed here.

Two sets of data arrays are used in the programs: one set contains information about the orders and the routes to which they belong; the other set contains

* REACT II Computer Program User's Manual, by Donna E. Clark and Michael Gray, DTNSRDC - 78/055, November 1978.

information about the ships used. In both cases the information contained in the arrays is initially stored in a particular sequence. Later the same information is used in a different sequence. For example, the orders generated by SOS are stored in the sequence in which they are input at the remote terminal. They are then scanned repeatedly and assembled into the final ship schedules.

The problem is how to re-organize the data in these arrays from their initial sequence to their final sequence. The first and most natural method is actual physical re-organization of the data. The advantage of this method is that the final arrays are easy to process, either by computer or the human mind. For example, the orders processed by ship #1 would appear first in the final arrays and would appear in the order in which the ship would service them. This physical re-organization can take place in two ways: through the use of duplicate arrays or by sorting the original arrays.

In using duplicate arrays the first set of arrays is examined and the appropriate element is selected, stored in the second set of arrays, and deleted from (or marked as processed in) the original arrays. The obvious disadvantage to this method, particularly when large amounts of data are being processed, is that the memory requirements of the program are doubled.

The second method of physically re-organizing data is by sorting. The initial arrays are examined, the chosen element is selected and physically moved to the first position in the arrays, and the remaining items in the arrays are shifted to make room for it. This process eliminates the need for duplicate arays and their large memory requirements; however, sorting is a time consuming process when the arrays involved are large.

Both these methods for physical re-organization of data were considered, but the constraints of time and space made them unacceptable.

A common method of processing large amounts of data is that of embedded links; this technique is used in several of the large data base management systems now commercially available. In this technique a sequence of data items in a large set of data arrays is linked together by providing an array of pointers or links. The pointer associated with a data element gives the address of the next datum in the sequence. A pointer external to the arrays gives the address of the first element in the sequence, the link variable associated with that element gives the address of the second element in the sequence, etc. The time constraints of sorting, where n^2 movements are required to sort n elements, are not encountered. The introduction of an additional array of pointers does not usually involve a significant increase in storage, since the data elements being sorted are usually made up of corresponding components of many parallel arrays (or the addresses of indices in the pointer array are much smaller than the items which they label).

The three methods of data restructuring are illustrated in Figures 2, 3, and 4.

A major advantage to the linked list method of data organization is that it speeds access to the data. For example, rather than searching an entire array for an item which is in a specific route, only the items in the route need be examined. The data examination process is made more efficient in SOS because there is a separate linked list for unprocessed orders, i.e., those orders not yet assigned to a ship route. As routes are built, orders pass from the unprocessed linked list to a specific ship's linked list. Thus each successive search of the unprocessed orders takes less time.

The savings in space and time of the linked list system must be paid for by increased complexity of the program code.

A separate linked list must be maintained if the arrays are to be searched in reverse order, or if items are to be inserted in a list. Thus two link arrays must usually be specified to determine a linear chain of items.

DATA PACKING

Because the size of orders and number of ports is limited, it was felt that each order entry could be placed, or "packed", in one data location. The array INFO represents all order information, and each entry has the following format:

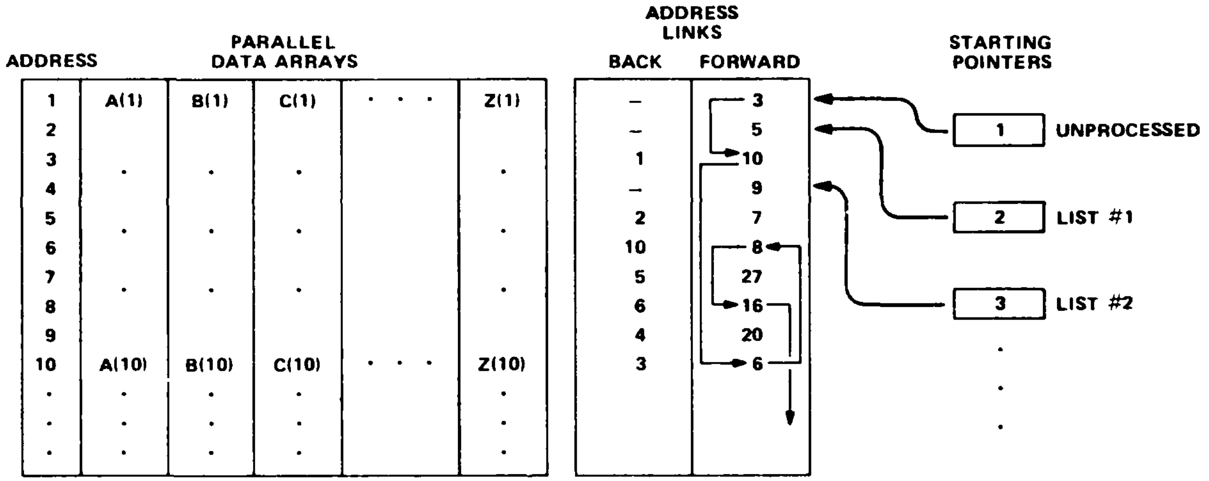
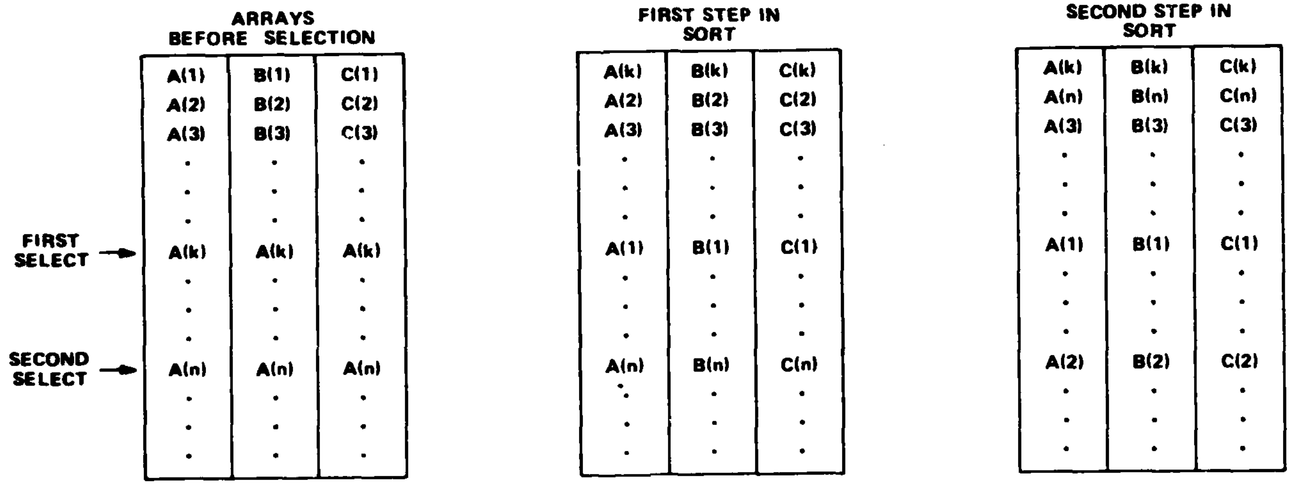


Figure 2 - Methods of Data Storage, Duplicate Arrays



ARRAYS BEFORE SELECTION

	A(1)	B(1)	C(1)
	A(2)	B(2)	C(2)
	A(3)	B(3)	C(3)
	.	.	.
	.	.	.
	.	.	.
FIRST SELECT →	A(k)	A(k)	A(k)
	.	.	.
	.	.	.
	.	.	.
SECOND SELECT →	A(n)	A(n)	A(n)
	.	.	.
	.	.	.
	.	.	.

FIRST STEP IN SORT

A(k)	B(k)	C(k)
A(2)	B(2)	C(2)
A(3)	B(3)	C(3)
.	.	.
.	.	.
.	.	.
A(1)	B(1)	C(1)
.	.	.
.	.	.
.	.	.
A(n)	B(n)	C(n)
.	.	.
.	.	.
.	.	.

SECOND STEP IN SORT

A(k)	B(k)	C(k)
A(n)	B(n)	C(n)
A(3)	B(3)	C(3)
.	.	.
.	.	.
.	.	.
A(1)	B(1)	C(1)
.	.	.
.	.	.
.	.	.
A(2)	B(2)	C(2)
.	.	.
.	.	.
.	.	.

Figure 3 - Methods of Data Storage, Sequential Sort

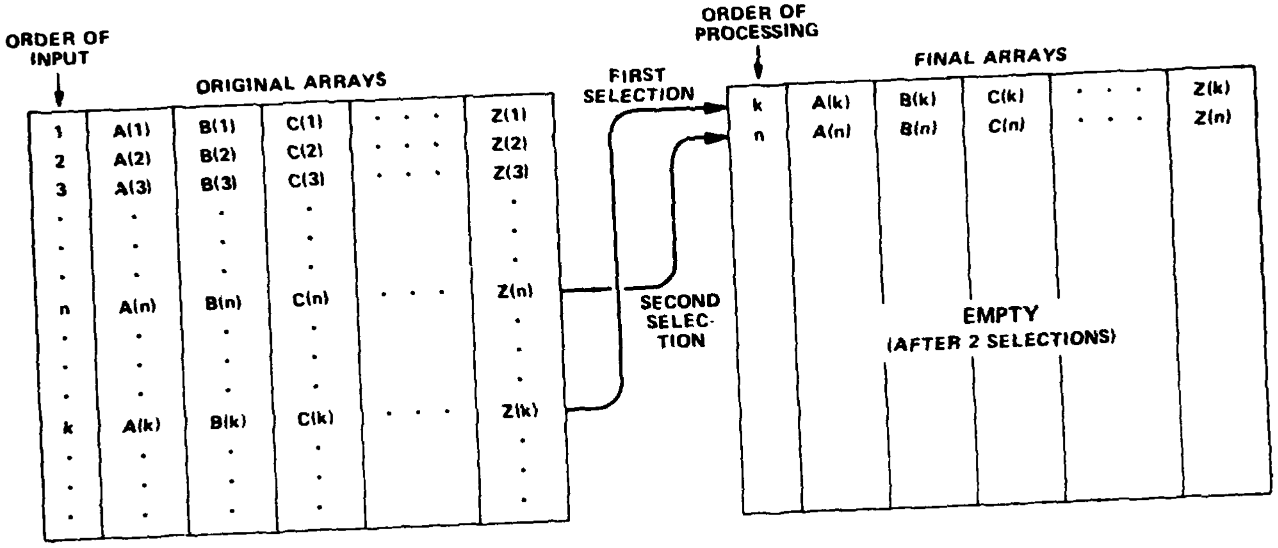


Figure 4 - Methods of Data Storage, Linked Lists

INFO Word Configuration:

Order allocation indicator +, unassigned -, assigned	Cargo type	Origin port number	Destination port number	Order size
1 digit	1 digit	2 digits	2 digits	7 digits

Data packing also reduces the number of internal sorts by listing one data element rather than four.

INPUT AND OUTPUT

When procedures specified in this User's Manual are followed, the data input to SOS is accomplished interactively from a remote printer terminal. The user must enter the following data:

Orders. List order sizes, cargo type(s), and origin and destination ports. Entries are made in free format. Data correctness messages are displayed on the terminal. Table 1 gives a brief description of each cargo type.

Ships. List ship types, home ports, and start times (days). Table 2 gives ship types and their characteristics.

Begin Time. Enter beginning times (days) for the schedules.

Route Length. Enter a maximum value for route duration in days. The default value is 100 days.

The Input Routine, VIP, checks the input data for validity. An order's origin and destination ports must match a built-in list of port names. Input numeric data (e.g., order size, times, ship's start time, and route duration) must be within specified ranges. The corrected data are made available to SOS through the Tape3 file. After all input has been entered satisfactorily, VIP will execute SOS.

TABLE 1 - CARGO TYPES

Cargo Type	Description
1	General Cargo
2	Special (Heavy lift)
3	RoRo
4	POL
5	Aircraft
6	Ammunition
7	Containers
8	Containers (Heavy lift)
9	Boats

TABLE 2 - SHIP TYPES AND THEIR CHARACTERISTICS

Ship Type	Ship Type Name	Capacity Measurement Tons	Speed Knots	Cargo Types Carried
1	BBS	10286	17.0	1,6,3,2,5
2	BBF	11729	20.7	1,6,3,2,5,7,8
3	ROROS	11803	17.4	3,2,5
4	ROROF	21905	23.5	3,2,5,7,8
5	SSCS	21106	17.4	7,8
6	SSCF	18200	21.1	7,8
7	NSSCS	18311	17.7	7,8
8	MSSCF	37544	23.2	7,8
9	COMBS	16721	17.4	7,8,1,3
10	COMBF	23311	23.6	1,3,2,5,7,8
11	LASH	29907	23.6	1,6,3,2,5,7,8,9
12	STRHV	12786	21.9	1,6,3,2,5
13	STRNSS	13584	16.0	1,6,3,2,5
14	SEABEE	31800	20.1	1,6,3,2,5,7,8,9

INPUT - INTERACTIVE DATA ENTRY

There are four distinct phases in the execution of SOS:

(1) Terminal and program call-up procedures

(2) Data entry and validation

(3) Schedule generation and output

(4) Schedule verification and storage

These phases are described in this section and a sample printer terminal session is given to illustrate the use of the scheduler.

Terminal and Program Call-up Procedures

The procedures for connecting the printer terminal by phone to the CDC 6600 at DTNSRDC, Carderock, MD are fixed by the system software and must be followed exactly. These procedures are quite straightforward, and the system will flag any errors for the user to correct.

Dial-Up Procedure. Although terminals may vary slightly in their configuration, the same dial-up procedure is applicable to all. The user dials the CDC 6600 at 202-227-3000 and listens for a high-pitched, continuous data tone. He then places the telephone hand-piece into the acoustic coupler at the back (typically) of the terminal and depresses the <RETURN> [*] key to signal the computer that he is ready to log in on the system.

Login Procedure. The computer sends the following information to the terminal:

```
DTNSRDC 5500 INTERCOM V 4.6
DATE  2/19/81
TIME 09.00.00
```

The user must then identify himself using secure codes (User ID and Access Number) supplied by the Users' Services Branch, Code 189, DTNSRDC. He enters:

```
<LOGIN,XXXXXXXXXX,SUP> <RETURN>
```

The Xs must be replaced by his valid user ID. SUP suppresses the system status messages.

[*] The symbols, < >, indicate user action or enclose information to be entered by the user.

If this entry is correct, the computer responds with:

```
        QQQQQQQQQQ ENTER ACCESS NUMBER-
```

The print head is positioned in the area shown here by Q's. This area is actually overstruck several times to protect the user's access number.

If either the user ID or access number is wrong, the system will reply with a message and request the user to repeat the input. After three unsuccessful tries the system locks the user out and suggests that he seek help from User Services.

If both user ID and access number are entered correctly, the system asks for instructions by issuing a prompt:

```
        COMMAND-
```

The user then proceeds with the next step of program call-up and initiation.

Program Call-Up and Initiation. After a successful login, the user prepares to run SOS by keying the following commands:

PROMPT	USER's COMMANDS	COMMENTS
COMMAND	<FETCH,SOSINQ>	Makes SOS available for terminal use.
COMMAND	<CONNECT,INPUT,OUTPUT>	Allows data entry and printout at terminal.
COMMAND	<SOSINQ>	Executes program

SOS now takes control of the terminal from the system and data are requested as outlined in the next section. An actual login and call-up session is given in the sample run.

Data Entry and Validation

After Login and Call-up SOS prints the following message:

```
            WELCOME TO SOS, SHIP ON-LINE SCHEDULING
                 TODAY'S DATE IS 02/19/81
            IF YOU WANT TO SEE A MENU, KEY 'M', IF NOT, KEY ' S'
---->
```

Entering anything but an M will suppress the menu listing. The entry of an M followed by a <RETURN> produces a list of possible SOS entries, some of which are optional.

Here is the menu:

```
----> M
```

```
                     SELECT OPTIONS FROM THE FOLLOWING MENU:
       O --- ORDERS
       V --- VEHICLES
       D --- DATE (IF DIFFERENT FROM TODAY'S)
       B --- TIME AT WHICH ROUTES WILL BEGIN
       L --- MAX LENGTH OF ROUTES (ALL VEHICLES)
       T --- TERMINATE RUN
      BU --- SPECIAL ORDER BUMP OPTION
      DI --- EVEN SHIP/LOAD DISTRIBUTION
       E --- END OF DATA ENTRY; GENERATE SCHEDULES
      SP --- SCHEDULE PRINT OPTION
      LI --- LIST, FOLLOWED BY 'O' FOR ORDERS, "V" FOR VEHICLES
             FULL LIST OF ORDERS IS GIVEN BY 'LI O ALL'
             PARTIAL LIST OF ORDERS IS GIVEN BY:
             LI O N1 N2  - LISTS ORDER N1 THRU N2
             LI O LAST N - LISTS LAST N ORDERS
             LI O N      - LISTS NTH ORDER
       R --- REMOVE FOLLOWED BY O or 'V' AND A NUMBER

SHIP TYPES AVAILABLE
      BBS
      BBF
      ROROS
      ROROF
      SSCS
      SSCF
      NSSOS
      NSSOF
      COMBS
      COMBF
      LASH
      STRHV
      STRNSS
      SEABEE
--->
```

These options are described individually in the paragraphs that follow. The items selected may be entered in any order. Note that the program uses only the first few letters to discriminate among entries. However, longer entries (e.g., ORDERS instead of O and LIST instead of LI) may be used.

'O' Option (Orders). The format for entering orders follows procedures now in use by REACT. The format is:

```
<ORIGIN> <SIZE> <DEST> <GEN TIME> <TYPE> <RETURN>
```

ORIGIN is the port of origin of the order

SIZE is the number of measurement tons to be moved

DEST is the destination port

GEN TIME is the time (days) when cargo is available at the origin port

TYPE is the cargo type (1 to 9)

An example of order input is:

```
---> O
INPUT ORDERS
   ORIG SIZE DEST GEN TIME TYPE
---> 1  4545   6  10    4   3
---> 1  3456  10   2    6   1
---> 3  3567  10   5    1   1
---> 4  6789  10   4    5   3
```

Only valid orders are accepted by the program. The following error messages may result from incorrect entries:*

```
UNRECOGNIZED ORIG ---------> [name]
UNRECOGNIZED DESTINATION --> [name]
SIZE MUST BE NUMERIC------>  [value]
MAX ORDER IS  9999999 MT-->  [value]
ORIGIN AND DESTINATION ARE THE SAME --> [name]
```

These error messages are self-explanatory. Incorrect data are ignored by the program, but the intended entries must be made again.

* The symbols, [ ], enclose information supplied by the SOS program.

'V' Option (Vehicles). SOS can accommodate the 14 types of vehicles/ships listed on the Menu. The user must now specify how many of each type of ship will be used in the scheduling algorithm, and specify each ship's home (initial) port and start start time (days). He need not specify every ship in the transportation fleet; SOS may be used to schedule only a portion of the total set of vehicles.

To begin vehicle specification, the user enters V and the program responds with:

```
INPUT VEHICLES
NO.  TYPE HOME PORT START TIME
```

A vehicle entry is as follows:

```
<n> <type> <home port> <start time> <return>
```

where n is the number of ships having the listed characteristics, type is the mnemonic code for the vehicles in the list, home port is the mnemonic code for the ship's initial port, and start time is the starting time (days) for the vehicles in this entry.

Examples of valid vehicle entries:

```
a)  2 BBF   1 1
b)  ROROF   2 1
c)  ROROS   3 2
```

Several vehicle entries may be entered on the same line, but they must be separated by a slash (/).

Example: `ROROF 1 1/2 BBF 2 7`

SOS does considerable data validation on the vehicle entries; errors in the vehicle data are much more consequential to the algorithm than errors in the other entries. Common typographical errors may be answered by one of these messages:

```
NUMERIC INPUT EXPECTED---> [bad entry]
VEHICLE ENTRY SKIPPED BECAUSE OF BAD DATA
```

SOS also checks the total number of vehicles of a type requested by the user against a maximum built into the program. When this maximum is exceeded, a message:

```
TOO MANY [xxxxxxx] THERE ARE [nn] LEFT
[mm] OF THE REQUESTED VEHICLE HAVE BEEN IGNORED
```

is printed. xxxxxxx is the type of vehicle, and nn + mm equals the number requested.

‘D’ Option (Date). The program obtains today’s date from the computer system, so this entry need be made only if the schedules are to be generated for some date other than the current date. *The user enters:* D *or* DA *or* DATE, followed by a <RETURN>, informing the program that this is the date to be printed on the schedules. The program responds with a prompt (--->) and the user then enters a date in any of the following forms:

a) DEC 18, 1978 or DECEMBER 18 78 etc.

b) 12/18/78 or 121878 etc.

c) 18 DEC 78 or 18 DEC 1978 or 18 DECEMBER 78 etc.

If an error is made, the program prints out:

```
    DATE NOT VALID, TRY AGAIN
--->
```

Only the last entered date is retained by SOS.

‘B’ Option (Begin Time). This option is used to specify the time at which the schedules will start. The user enters B or BEGIN and a <RETURN>; the program responds:

```
    ENTER BEGINNING TIME FOR ROUTES
    --->
```

The user then enters a valid start time (days). Any error in input - e.g., alphabetic characters, etc. - results in the following message:

```
      ERROR IN INPUT = [entry], TRY AGAIN
```

The corrected data can be entered at any time by requesting the ′B′ option .

‘L’ Option (Length). SOS builds vehicle (ship) routes in accordance with the length of shipment time (days) determined by the user.

```
    User enters:  L
    Program replies:  ENTER MAX LENGTH OF ROUTE FOR ALL VEHICLES
                      --->
```

The user then enters the route length in days. Any error in input - e.g., alphabetic characters, etc., - results in the following message:

```
    ERROR IN INPUT = [value], TRY AGAIN
```

The corrected data can be entered at any time by requesting the ‘L’ option.

Run Control Options

The following options control the execution of the SOS program.

'T' Option (Terminate Run). This option is used when the user wishes to terminate data entry and abort the related SOS run. Upon entry, control is given back to the operating system.

'BU' Option (Bump). Orders entered under this option are given priority and are scheduled ahead of other orders not yet scheduled. (Not implemented).

'DI' Option (Distribute). This option directs the SOS scheduling algorithm to distribute orders evenly among the available ships. When DI is engaged, program option = 2; otherwise, this option = 1 by default.

Input/Output Options

The following options enable the user to list and update the current input parameters. A provision is also made to allow the user to print the ship's schedules.

'E' Option (End of Data Entry). Upon entry of this option, the program prints a summary of the data entered.

Example:

```
DATA ENTRY SUMMARY:
  DATE       = 21881
  BEGIN TIME =     3
  MAX ROUTE  =   120 DAYS

  3 BBF
  2 ROROS
  1 COMBF
  4 ORDERS,                  60268 MTS
IF INPUT IS NOT COMPLETE, TYPE 'MORE'
OTHERWISE, TYPE ANYTHING ELSE------->
```

If the data are accurate, any entry will start the next phase of the program, schedule generation and output. If an error or inconsistency shows up in the summary, typing MORE will allow the user to go back and make corrections or additional entries.

The user should check carefully that the following minimal information has been entered:

a) begin time

b) at least one order

c) at least one vehicle

When the data appear correct to the user, he enters GO and the scheduling begins.

'SP' Option (Schedule Output). Unless specified by 'SP' the ship schedules will be suppressed.

'LI' Option (List). Some of the errors commonly made in specifying the orders and vehicles to be used will look valid to SOS. For example, an order originating at port 3 may accidentally be entered as 13, which is also a valid location to SOS. Errors of this type made in the date, length, or begin time (D, L, or B options) are easily corrected by entering the correct information. Vehicle and order entries, however, are lists and a means must be provided for specifying which element in the list is to be corrected. This provision is made in the LI (LIST) and R (REMOVE) commands.

The LI command gives a printout of either the vehicles or orders which have aleady been entered. The format LI V lists the vehicles, their home ports, and their start times along with a reference number as shown in the following example:

```
LI V:    VEHICLES
         (Ref No.)    (Type)    (Home port)    (Time, days)
             1         BBF           5               2
             2         BBF           5               2
             3         BBS          12               2
             4         SCSS         14               1
             5         SCSS          7               2
```

The analogous entry for listing orders is LI 0, which prints out the orders listing a reference number, size, origin, destination, generation time, and cargo type for each order. Since the listing of orders can be rather time consuming, a partial list feature for orders is available. The valid forms are these:

LI 0 nn mm	list orders nn through mm, where nn and mm are the order list limits
LI 0 nn	list order nn, where nn is the order reference number
LI 0 LAST	list last order
LI 0 LAST mm	list last mm orders, where mm is the number of orders to be listed
LI 0 ALL	list all orders

'R' Option (Remove). Used in conjunction with the LI, 0, and V options, the R option allows for the selective correction of order and vehicle entries. The formats are

R 0 nn and

R V nn

where nn is the reference number supplied by the LI option. The (nn) the entry is deleted from the list and the successive entries are renumbered to close the gap. Because of this renumbering it is wise to remove orders and vehicles in reverse numerical sequence (higher numbers first) to avoid inadvertantly deleting the wrong entry. Once the erroneous entries have been removed from the list, correct data can be re-entered using the V or 0 commands.

Schedule Generation and Output

This phase of the program requires no action on the part of the user. Schedules are generated by the SOS algorithm and the results are printed. A sample printout of SOS, along with the Login, Call-up, and Data Entry phases, is given in the sample run section. The output from SOS comprises a complete input data summary and the various vehicle schedules which have been generated.

Two possible errors may be flagged by SOS or the system in this phase:

1. If there are too many orders for the requested vehicles to handle, the program responds with a message such as:

```
------------------------------------------------------------------------
THE FOLLOWING ORDERS (OR PARTS) COULD NOT BE FILLED
------------------------------------------------------------------------
                    . . .
[nn] MTS OF ORDER [mm] FROM [orig] TO [dest]
                    . . .
```

Here, nn is the number of undelivered measurement tons, mm is a number given to the order during data entry and orig and dest refer to the origin and destination ports of that order.

2. If a relatively large number of orders has been entered (say 75-100), the program may run out of computer time. When this occurs, the system will print out the message:

```
CP TIME LIMIT EXCEEDED.
```

The program output, if any, is lost and the user must start over at the data entry phase.

Schedule Verification and Storage

An example of SOS schedules is shown in the next section. The user should examine the output carefully to see that there are no problems. If he does not like some aspect of the schedules, they may be modified manually. On occasion there may be a problem which can be remedied by re-generating the schedules. After the last table has been printed, the terminal control returns to the system, which requests its next instruction with the usual prompt:

```
COMMAND-
```

to which the user responds:

```
<LOGOUT>
```

The system will print out a summary of the time and charges for this terminal session, and the user can then disconnect the phone and turn off the terminal.

OUTPUT - SAMPLE SESSION

Schedule output from SOS consists of a summary of the input data, the optional schedules for the individual ships, and the cargo shipment tables. Each schedule gives the ship name, route starting time, and dates in a header; a list of scheduled stops specifying port, time, measurement tons picked up or delivered, order reference number, cargo type, and approximate stay time at the port; and a trailer of finishing time and port, time still available, and number of measurement tons moved. SOS also creates a system schedule file, TAPE3, which is used to generate the cargo shipment tables. The cargo shipment tables identify cargo movement with respect to ship type and origin and destination ports.

```
    WELCOME TO SOS, SHIP ON-LINE SCHEDULING

         TODAY'S DATE IS 11/03/81

    IF YOU WANT TO SEE A MENU, KEY 'M', IF NOT KEY'S'

--->s
--->b

ENTER BEGINNING TIME FOR ROUTES
--->3
--->l

ENTER MAX LENGTH OF ROUTE FOR ALL VEHICLES
--->100
--->sp
SCHEDULE PRINT OPTION ENGAGED
--->di
ROUTE SPLIT OPTION ENGAGED
--->o

INPUT ORDERS
ORIG  SIZE  DEST  GEN TIME  TYPE
--->1 33445 13 2 1/24567 12 1 3
--->li o all
       1   33445 1      13      1    2.
       2   24567 1      12      3    1.
--->v

INPUT VEHICLES
NO.  TYPE  HOME PORT  START TIME
--->2 bbf 2 3
--->e


DATA ENTRY SUMMARY :

    DATE      = 110381
    BEGIN TIME =     3
    MAX ROUTE =    100 DAYS


     2 BBF
     2ORDER,         58012 MTS
 IF INPUT IS NOT COMPLETE, TYPE 'MORE'
    IF NOT, TYPE ANYTHING ELSE ------->go


    THIS CONCLUDES THE DATA ENTRY FOR SOS
    SCHEDULES WILL BE PRINTED SOON; BE PATIENT
```

```
                              ---------------------------
                              SOS REGULAR ORDER PROGRAM
                              ---------------------------


                                    11/ 3/81

                                       3.0

                                     (OPT=2)

                                     ------
                                     ORDERS
                                     ------

          1   33445   MTS FROM 1      TO 13    TYPE  1 GEN TIME =     2.
          2   24567   MTS FROM 1      TO 12    TYPE  3 GEN TIME =     1.

                                 -----------------
                                 VEHICLES SELECTED
                                 -----------------

               1 VEHICLE BBF     1, HOME PORT 2       ,START TIME =    3.
               2 VEHICLE BBF     2, HOME PORT 2       ,START TIME =    3.


            SHIP PROCESSING

      LAST SHIP PROCESSED BBF        1 AT 2      TIME LEFT=  47.
      LAST SHIP PROCESSED BBF        2 AT 2      TIME LEFT=  47.
      RESET PHASE IN PROCESS
      LAST SHIP PROCESSED BBF        1 AT 2      TIME LEFT=  97.
      LAST SHIP PROCESSED BBF        1 AT 13     TIME LEFT=  73.
      LAST SHIP PROCESSED BBF        1 AT 13     TIME LEFT=  48.
      LAST SHIP PROCESSED BBF        1 AT 1      TIME LEFT=  26.
      LAST SHIP PROCESSED BBF        1 AT 12     TIME LEFT=  22.
      LAST SHIP PROCESSED BBF        2 AT 2      TIME LEFT=  97.
      LAST SHIP PROCESSED BBF        2 AT 1      TIME LEFT=  82.
```

```
                   SHIP - BBF       1
                   START TIME -    3.
                   DATE         110381


***********************************************************************
     STOP  SITE   TIME     DELIVER       PICK UP     ORDER   STAY TIME
***********************************************************************

        1 1          6                  11729 MTS  1     1      8 *SPLIT
              $ BERTHING DELAYS ENCOUNTERED $  3.93 DAYS
        2 13        22    11729 MTS  1                   1      8 *SPLIT
        3 1         34                  11729 MTS  1     1      8 *SPLIT
        4 13        47    11729 MTS  1                   1      8 *SPLIT
        5 1         59                   9987 MTS  1     1      7 *SPLIT
                                         1742 MTS  3     2      1 *SPLIT
        6 13        70     9987 MTS  1                   1      7 *SPLIT
        7 12        83     1742 MTS  3                   2      2 *SPLIT
        8 1         94                  11729 MTS  3     2      4 *SPLIT
        9 12       107    11729 MTS  3                   2      4 *SPLIT

                   ROUTE ENDED
                   LOCATION =2
                   TIME =  119
                   NO MTS MOVED =         46916



                   SHIP - BBF       2
                   START TIME -    3.
                   DATE         110381


***********************************************************************
     STOP  SITE   TIME     DELIVER       PICK UP     ORDER   STAY TIME
***********************************************************************

        1 1          6                  11096 MTS  3     2      3 *SPLIT
        2 12        18    11096 MTS  3                   2      3 *SPLIT

                   ROUTE ENDED
                   LOCATION =2
                   TIME =   31
                   NO MTS MOVED =         11096
     SHIP    NUMBER  NUMBER  CARGO (MTS)  AVG ROUTE     AVG PERCENT
     TYPE     AVAIL   USED     MOVED      LENGTH-DAYS   UTILIZATION

  2 BBF          2       2        58012           69             97


TOTAL            2       2        58012

     PORT     CARGO (MTS)    CARGO (MTS)    CARGO (MTS)    CARGO (MTS)
     NAME        AVAIL       SHIPPED FRM    DELIVERED TO    UNMOVED
  1  1            58012          58012              0              0
 12 12                0              0          24567              0
 13 13                0              0          33445              0


TOTAL             58012          58012          58012              0


    BERTHING DELAYS SUMMARY

         PORT      SHIPS       AVG TIME
         NAME      QUEUE     QUEUED (DAYS)


      1  1             0             0.0
      2  2             0             0.0
      3  3             0             0.0
      4  4             0             0.0
      5  5             0             0.0
      6  6             0             0.0
      7  7             0             0.0
      8  8             0             0.0
      9  9             0             0.0
     10 10             0             0.0
     11 11             0             0.0
     12 12             0             0.0
     13 13             1             3.9
      STOP
      035600 MAXIMUM EXECUTION FL.
       1.450 CP SECONDS EXECUTION TIME.
COMMAND- logout
CPA      8.945 SEC
SS       9.738 SEC
EST. SYSTEM COST $     .99
EST. CONNECT COST $    2.08
CONNECT TIME    0 HRS.   25 MIN.
 11/03/81  LOGGED OUT AT 14.16.30.
       <
```

INITIAL DISTRIBUTION

Copies	
7	OPNAV 405
	2 Mr. B. Gruber
	1 Capt. H. Spencer
	1 CDR J. Gunnon
	1 Code 40 RADM. Cruden
	1 Code 404 Mr. Presten
	1 Code 608 CDR F. Stout
1	CINCLANTFLT N-4
1	PACFLT J-4
1	CINCLANT LCDR D. Wentworth
1	CINCUSNAVEUR LT. R. Binn
1	CINCPAC Mr. J. Feir
2	DTIC
2	DLSIE
1	JDA
1	RDF
1	MCDEC
1	NAVSUP Code 05 Mr. H. Lieberman
2	NAVMAT
	1 Code 043 Mr. R. R. Hallmark
	1 Code 01 Mr. J. T. Kammerer
1	MARAD Mr. Frank Case

CENTER DISTRIBUTION

Copies	Code	Name
1	18	G.H. Gleissner
2	1809.3	D. Harris
1	185	T. Corin
1	187	M. Zubkoff
1	187	R. Ploe
1	187	C. Emberger
1	1871	H. Applegate
20	1871	R. Melton
1	1871	J. St. Laurent
1	1872	P. Friedenberg
1		LTCOL K. Thompson
10	5211.1	Reports Distribution
1	522.1	Unclassified Lib (C)
1	522.1	Unclassified Lib (A)
1	93	L. March